



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/547,539	04/12/2000	Michael D. Smith	144243.1/40062.53US01	5841
23552	7590	12/30/2004	EXAMINER	
MERCHANT & GOULD PC P.O. BOX 2903 MINNEAPOLIS, MN 55402-0903			VU, TUAN A	
			ART UNIT	PAPER NUMBER
			2124	

DATE MAILED: 12/30/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 09/547,539	Applicant(s) SMITH ET AL.	
	Examiner Tuan A Vu	Art Unit 2124	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
 - If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
 - If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
 - Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 23 August 2004.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-6, 10-29, 33-39, 43-52, 56-61 and 65-68 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-6, 10-29, 33-39, 43-52, 56-61 and 65-68 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. This action is responsive to the Applicant's response filed 8/23/2004.

As indicated in Applicant's response, claims 1, 19, 27, 34 and 56 have been amended; and claims 7-9, 30-32, 40-42, 53-55, 62-64 canceled. Claims 1-6, 10-29, 33-39, 43-52, 56-61, and 65-68 are pending in the office action.

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

3. Claims 1-6, 10-29, 33-39, 43-52, 56-61, and 65-68 are rejected under 35 U.S.C. 103(a) as being unpatentable over Ishizuka et al., USPN: 5,313,635 (hereinafter Ishizuka), in view of Woolsey et al., USPN: 6,029,000(hereinafter Woolsey); and further in view of Liu et al., USPN: 6,058,482 (hereinafter Liu).

As per claim 1, Ishizuka discloses a compiling system having method steps of:

transmitting compilation information from a first subsystem to a second subsystem (col. 2, lines 42-62);

compiling computer program code on the second subsystem based on the compilation information received from the first subsystem (col. 2, lines 62-68; col. 5, lines 17-30; Fig. 7), including at least compilation instructions related to the particular machine-executable code required by the first subsystem (e.g. steps 52, 54-61, Fig. 5b – Note: gathering information such as language name, compile command, source file, library-installed machine information, name of

Art Unit: 2124

client host is equivalent to instructions related to a specific machine code required by the first subsystem) to the second subsystem; and

receiving the compiled code from the second subsystem into the first subsystem (col. 3, lines 18-23).

But Ishizuka does not explicitly specify that the step of compiling in the second subsystem generates machine-executable code. However, Ishizuka's clearly shows that an object file is an executable format (S12/*a.out*, S11/*execution format: a.out* -- Fig. 8 -- Note: *a.out* is the default output file from running an executable like C programs) when Ishizuka discloses returning an object file to the requesting client (e.g. col. 3, lines 14-19; Fig. 5a-b) after collecting all machine-specific information (Fig. 5a) to generate such object file to alleviate burden on the requesting client machine (e.g. col. 1, line 53 to col. 2, line 21). Hence, it is noted that Ishizuka has at worst hinted that the code delivered is a machine-executable form. Besides, gathering libraries to generate object modules for linking into an executable is a known concept at the time the invention was made; and the concept of binding libraries with object code is taught by Ishizuka (col. 7, lines 16-22). Just in case the delivered object file from Ishizuka's compiler-installed server has not already been produced in a machine-executable form, it would have been obvious for one of ordinary skill in the art at the time the invention was made to adjust Ishizuka's method of compilation so that the generated object file as disclosed would be linked into a machine-executable based on the information provided by the first subsystem because this would yield a code in ready form for execution/use by the first subsystem without additional resources usage from the part of the latter, hence improve time and resources efficiency as well as extensibility (Ishizuka: col. 2, lines 32-39; col. 7, lines 22-30).

Ishizuka does not disclose prior to receiving the machine executable code, the step of detecting whether the 1st and 2nd subsystems are connected by a secure link so to determine whether the second subsystem is a trusted source, using a receipt policy. By the time the invention was made, electronic content or software being distributed over corporate network or internet; and being subject to integrity/security check or compliance verification against enterprise/business policies/rules in terms of computer data reception/acceptance was a known concept of network distribution or transmission, e.g. a web-based communications secure link. Woolsey, in a similar system having method of using a cross-compiler for generating executable code, or native code, by a second subsystem for a first subsystem analogous to Ishizuka's method, teaches a detection process for checking if the second subsystem is a trusted source prior to receiving the executable into the first subsystem (col. 20, line 50 to col. 21, line 17; Fig. 6), i.e. Woolsey's concept of establishing whether a sender is a trusted source entails a policy as to whether or not to accept as taught by known concept in regard to secure a link for data acceptance. Liu, in a similar system having method of using a network communication link for providing Web/Java associated executable code similar to Woolsey's method by a second subsystem for a first subsystem also analogous to Ishizuka's method, teaches policies implemented to ensure that the source provider is a trusted source prior to receiving the executable into the first subsystem (col. 3, line 55 to col. 4, line 3; Fig. 1; col. 9, lines 39 to col. 10, line 24; Fig. 5 – Note: verifying references of source for trust checking reads on connection ensured via secure link). Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to add to Ishizuka's compiling method both the detection of trusted sources of received executables as taught by Woolsey in order to secure transactions

Art Unit: 2124

with safer acquisitions of data between the subsystems involved therein; and enhancing Woolsey's teaching with methods to enforce a policy for receiving data via secure links and receipt of executables following the policy of receipt as taught by Liu because this allows more secure transactions or safer acquisitions of data between the subsystems involved therein; and provides added trust and security in the access or execution of programs resulting from those transactions in those subsystems' local environment, especially when data hacking or spoofing has increasingly become a threat (Liu: col. 3, lines 41-60)

As per claim 2, Ishizuka further discloses in the method of claim 1 that the step of transmitting compilation information includes transmitting compilation information from a first subsystem to a second subsystem in response to a request to compile computer program code into machine-executable code (col. 2, line 59 to col. 3, line 14; Fig. 7).

As per claim 3, in method of claim 1 above Ishizuka further discloses the step of transmitting of compilation information by the first subsystem to the second subsystem (col. 3, lines 9-14; col. 4, lines 54-60; Fig. 7); but does not disclose that such transmitted compilation information is in intermediate language code. Official notice is taken that, at the time the invention was made, the use of intermediate code, e.g. Java bytecodes, as a platform-independent software program form to be transmitted across multi-host environments/networks for machine-specific compilation into executable code is a well-known concept. Such practice about transferring code as intermediate form was evidenced by the use of bytecode as taught by Liu (col. 3, lines 8-30). Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to modify Ishizuka's step of transmitting compiling information so that, instead of just providing platform-specific information and program-related source and

Art Unit: 2124

libraries data, the well-known intermediate form (also taught by Liu) of program code having cross-platform property therein would be transmitted, whenever available, by the first subsystem to the second subsystem for compilation because it would relieve both subsystems from allocating and storing machine-specific resources/utilities.

As per claim 4, Ishizuka in the offloading compilation method of claim 1 does not disclose that the step of transmitting compilation information includes transmitting compilation information from a small device to a second subsystem. However, Woolsey in a method step for downloading executables analogous to that of Ishizuka's method teaches the transmitting of compilation information being from a small device or first subsystem to the second subsystem (e.g. *host-DSP interface layer 62* – Fig. 2; *DSP-API* – col. 3, line 66 to col. 9, line 21; col. 1, line 18-28). This teaching hence is suggesting communication of compilation information to and from the 2 subsystems for the conversion and delivery of target code within the communication paradigm wherein the first subsystem is portable/embedded processor of a smaller device, e.g. DSP or PDA, in order for the latter to obtain the suitable native codes. Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to include into Ishizuka's system/method for providing executables to requesting client host machines the above transmission of compiling information from and reception of executables by a small device so taught by Woolsey because of the need to comply to increasing popularity and functionality of hand-held devices and the technological demands in communications involving the internet and embedded processors represented by those small devices, such demands being increasingly sought for at the time the invention was made, just as mentioned in Woolsey's background (*cellular phone, PDA* – col. 1, line 18-28).

As per claim 5, and with respect to the offloading compilation method of claim 4 above, Ishizuka does not disclose that the step of transmitting compilation information includes transmitting compilation information from a cellular phone to a second subsystem. However, Woolsey in a method step for downloading executables analogous to that of Ishizuka's method teaches the transmitting of compilation information from a cellular phone to the second subsystem (see Woolsey: *cellular phone, PDA* – col. 1, line 18-28). The same reasons used to reject claim 4 above by virtue of obviousness still apply here.

As per claim 6, Ishizuka discloses that the step compiling computer program code from claim 1 above includes compiling program code on the second subsystem based on the compilation information received from the first subsystem (e.g. col. 2, lines 42-58); but does not teach that such compilation step compiles intermediate code into machine-executable code. Official notice is taken that providing multi-platform code using an intermediate form was a known concept at the time of the invention. Such practice about transferring code as intermediate form was evidenced by the use of bytecode as taught by Liu (col. 3, lines 8-30). Hence, one of ordinary skill in the art at the time the invention was made would recognize the obvious implementation of a machine-executable as a result of the compiling of intermediate language code for cross-platform distribution/compilation such as taught by Liu; the rejections for which limitations have been set forth in claims 1 (machine-executable) and 3 (intermediate language), respectively, above. Therefore, it would have been obvious to provide this form of intermediate form as taught by Liu to the distribution of code as intended by Ishizuka for the reasons of cross-platform expediency and resource saving associated the use of intermediate code transmission.

As per claims 10, 11, and 12, Ishizuka does not expressly disclose in the method in claim 1 above the using of the machine-executable code in the first subsystem (re claim 10); such using step includes storing the machine-executable code on the first subsystem (re claim 11) and executing the machine-executable code on the first subsystem (re claim 12). However, in view of the admitted prior art as disclosed via col. 1, lines 13-29 and Fig. 10, it would have been obvious for one ordinary skill in the art at the time the invention to recognize the obvious anticipation of the above limitations in Ishizuka's system/method because the purpose of obtaining an machine-executable code by a subsystem from another subsystem is for the former being able to possess and use, i.e. obviating the need for further quest of, that code.

As per claims 13, and 14, Ishizuka according to the method of claim 1 discloses that the step of transmitting includes transmitting compilation information and computer program code from a first subsystem to a second subsystem (re claim 13 -- col. 3, lines 9-14; col. 4, lines 56-60); that before the step of compiling, the step of retrieving computer program code for compilation into machine-executable code (re claim 14 -- col. 5, line 51 to col. 6, line 7; Fig. 8).

As per claim 15, and in reference to claim 14 above, Ishizuka discloses before the step of compiling, a step of retrieving computer program code for compilation but does not teach that the step of retrieving includes retrieving program code from a third subsystem. However, Woolsey, in a similar system having method step of generating executable code by a second subsystem upon request from a first subsystem, teaches that the retrieving of program code for compilation in the second subsystem includes retrieval of computer program code from a seemingly unknown/untrusted third subsystem (e.g. col. 20, line 50 to col. 21, line 17; Fig. 5-6). Therefore, it would have been obvious for one of ordinary skill in the art at the time the

Art Unit: 2124

invention was made to include the step of retrieving program code/components by the second subsystem from a third subsystem as taught by Woolsey to Ishizuka's system because this will relieve the second subsystem from the burden of allocating large local resources for storing as many program codes/compilation information as needed to provide for all compilation requests from different platforms; thereby possibly improve the turn-around time for fulfilling requests from related components within a network.

As per claim 16, Ishizuka according to the method of claim 1 discloses that the step of compiling includes decoding the compilation information (col. 6, lines 18-32; col. 7, lines 6-22; Fig. 7).

As per claim 17, Ishizuka according to the method of claim 1 discloses that the step of transmitting compilation information from a first subsystem to a second subsystem includes transmitting compilation information from a first subsystem to a second subsystem wherein the first and second subsystems are components of a single system (Figs. 8-9).

As per claim 18, Ishizuka according to the method of claim 1 discloses that the step of transmitting includes transmitting compilation instructions (*compile command*) from a first subsystem to a second subsystem (e.g. col. 4, lines 30-44; col. 6, lines 36-44).

As per claim 19, Ishizuka discloses a first subsystem for compiling program code for execution in a second subsystem having method steps of:

receiving compilation information from the second subsystem (col. 2, lines 59-62; col. 3, lines 1-18), including at least compilation instructions related to the particular machine-executable code required by the second subsystem (e.g. steps 52, 54-61, Fig. 5b – Note: the

Art Unit: 2124

second subsystem gather the compiling instructions first before sending to the compiler-installed first subsystem) ;

compiling computer program code based on the compilation information received from the second subsystem (col. 2, lines 62-68; col. 5, lines 17-30; Fig. 7); and

transmitting the compiled code to the second subsystem (col. 3, lines 18-23).

But Ishizuka does not explicitly disclose that the first subsystem step of compiling code to be transmitted to the second subsystem generates machine-executable code. However, the rejection for which limitation has been set forth in claim 1 above.

Nor does Ishizuka disclose detecting whether the 1st and 2nd subsystems are connected by a secure link so to determine whether the second subsystem is a trusted source, using a receipt policy. But this limitation has been addressed in claim 1 above using Woolsey enhanced with Liu's teaching.

As per claim 20, and with respect to the method of compilation of claim 19 above, this claim includes the same step limitations as in claim 4. These limitations are rejected using the combination of Ishizuka and Woolsey as set forth in claim 4 above.

As per claim 21, and in reference to claim 20 above, this claim includes the same step limitations as in claim 5. These limitations are rejected using the combination of Ishizuka and Woolsey as set forth in claim 5 above.

As per claim 22, Ishizuka in the method of claim 19 above discloses that the compiling by the first subsystem is based on the compilation information received from the second subsystem (col. 5, lines 17-30) but does not teach that the step of compiling computer program code includes compiling intermediate language code into machine-executable code. However,

Art Unit: 2124

the rejections for which limitations have been set forth in claims 1 (machine-executable) and 3 (intermediate language), respectively, above. Therefore, it would have been obvious.

As per claims 23 and 24, Ishizuka discloses in the method of claim 19 above that the step of receiving includes receiving compilation information and computer program code from the second subsystem (re claim 23 -- col. 3, lines 9-14; col. 4, lines 56-60); and that the method further comprises before the step of compiling, retrieving computer program code for compilation into machine-executable code (re claim 24 -- col. 5, line 65 to col. 6, line 7; Fig. 8).

As per claim 25, Ishizuka discloses in the method of claim 24 above that the step of retrieving computer program code includes retrieving computer program code from the first subsystem (col. 5, line 51 to col. 6, line 7; Fig. 8).

As per claim 26, Ishizuka discloses in the method of claim 19 above that the step of compiling includes decoding the compilation information (col. 6, lines 18-32; col. 7, lines 6-22; Fig. 7).

As per claim 27, Ishizuka discloses a method in a second subsystem for offloading compilation to a first subsystem having a program code compiler, the method comprising:

transmitting compilation information to the first subsystem (col. 2, lines 42-62), including at least compilation instructions related to the particular machine-executable code required by the second subsystem to the first subsystem (e.g. steps 52, 54-61, Fig. 5b – Note: the second subsystem gather the compiling instructions first before sending to the compiler-installed first subsystem); and

receiving machine-executable code, compiled from the compilation information, from the first subsystem (e.g. Fig. 7; col. 2, lines 42-58; col. 3, lines 18-23).

But Ishizuka does not disclose detecting whether the 1st and 2nd subsystems are connected by a secure link so to determine whether the second subsystem is a trusted source, using a receipt policy. But this limitation has been addressed in claim 1 above using Woolsey enhanced with Liu's teaching.

As per claims 28 and 29, in reference to the method of claim 27 above, Ishizuka further discloses that the step of transmitting compilation information includes transmitting compilation information in response to a request to compile computer program code into machine-executable code (re claim 28 – col. 3, lines 9-14; Fig. 7); but does not teach that the step of transmitting compilation information includes transmitting compilation information written in intermediate language code (re claim 29). However, such limitation has been set forth and addressed in claim 3 above; therefore, it would have been obvious.

As per claim 33, in reference to the method of claim 27 above, Ishizuka further discloses that the step of transmitting includes transmitting compilation information and computer program code to a first subsystem (col. 3, lines 9-14; col. 4, lines 56-60).

As per claim 34, Ishizuka discloses a computer-readable medium for encoding a computer program for executing a computer process for offloading compilation, the computer process comprising the same process steps including

sending (from a first subsystem);

compiling (on the second subsystem); and

receiving (into the first subsystem) as recited in claim 1 above; therefore, the same rejection used in claim 1 still apply here.

But Ishizuka does not expressly disclose that the compiled code received by the second subsystem from the first subsystem is machine-executable code. However, such limitation has been set forth and addressed in claim 1 above.

Nor does Ishizuka disclose detecting whether the 1st and 2nd subsystems are connected by a secure link so to determine whether the second subsystem is a trusted source, using a receipt policy. But this limitation has been addressed in claim 1 above using Woolsey enhanced with Liu's teaching.

As per claims 35-39, Ishizuka discloses in the computer process of claim 34 the step of sending, or compiling as having the same limitations set forth in claims 2-6, respectively; therefore, the same rejections used in claims 2-6, respectively, are herein applied.

As per claims 43-47, Ishizuka discloses in the computer process of claim 34 the step of sending, or compiling as having the same limitations set forth in claims 13, 14, 15, 16, and 18 respectively; therefore, the same rejections used in claims 13, 14, 15, 16, and 18, respectively, are herein applied.

As per claim 48, Ishizuka discloses a system for offloading compilation, the apparatus comprising

a transmit module that transmits compilation information from a first subsystem to a second subsystem (col. 2, lines 42-62) including at least compilation instructions related to the particular machine-executable code required by the second subsystem to the first subsystem (e.g. steps 52, 54-61, Fig. 5b);

a compile module that compiles program code into machine-executable code on the second subsystem based on the compilation information received from the first subsystem (e.g.

Art Unit: 2124

Fig. 7; col. 2, lines 62-68; col. 5, lines 17-30); and

a receive module that receives the machine-executable code from the second subsystem into the first subsystem (e.g. col. 3, lines 18-23).

But Ishizuka does not explicitly disclose that the step of compiling in the second subsystem generates machine-executable code. However, such limitation has been set forth and addressed in claim 1 above; therefore, it would have been obvious.

Nor does Ishizuka disclose detecting whether the 1st and 2nd subsystems are connected by a secure link so to determine whether the second subsystem is a trusted source, using a receipt policy. But this limitation has been addressed in claim 1 above using Woolsey enhanced with Liu's teaching.

As per claim 49, in reference to the system of claim 48 Ishizuka further discloses the transmit module as having the same limitations as seen in claim 28 above; therefore, the same rejection used in claim 28 still applies here.

As per claim 50, in reference to the system of claim 48 Ishizuka further discloses the compilation information as having the same limitations as seen in claim 29 above; therefore, the same rejection used in claim 29 still applies here.

As per claims 51 and 52, in reference to the system of claim 48, these claims limitations correspond to claims 4 and 5 above, respectively and are rejected using the corresponding rationale using the combination of Ishizuka and Woolsey as set forth therein.

As per claim 56, Ishizuka discloses a computer system for executing a computer program, the computer program having instructions for executing a computer process for offloading compilation, the computer process comprising the same steps of

Art Unit: 2124

sending (from a first subsystem);
compiling (on the second subsystem); and
receiving (into the first subsystem) as recited in claim 34 above; therefore, the same rejection used against the respective limitations in claim 34 still apply here.

But Ishizuka does not disclose a computer data signal embodied in a carrier wave readable by a computing system and encoding program instructions to carry out the computer process as depicted above. However, one of ordinary skill in the art at the time the invention was made would recognize the need for a readable carrier wave medium for storing instructions to carry out the functionality of Ishizuka's system. Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to implement a computer-readable carrier wave to embody the computer data signal and encode the computer program instructions for executing the computer process disclosed by Ishizuka because it would facilitate a faster, more convenient and widespread distribution/dissemination of computer program signal encoding the program process instructions to/for the benefit of a wider pool, e.g. via the internet, of recipients intended for the use/access of such computer program/product.

Further, Ishizuka does not explicitly disclose that the step of compiling in the second subsystem generates machine-executable code. However, such limitation has been set forth and addressed in claim 1 above; therefore, it would have been obvious.

Nor does Ishizuka disclose detecting whether the 1st and 2nd subsystems are connected by a secure link so to determine whether the second subsystem is a trusted source, using a receipt policy. But this limitation has been addressed in claim 1 above using Woolsey enhanced with Liu's teaching.

Art Unit: 2124

As per claim 57, in reference to the system of claim 56 above, Ishizuka further discloses the step of sending as having the same limitations as claim 35; therefore, the same rejection used in claim 35 still applies here.

As per claim 58, in reference to the system of claim 57 above, Ishizuka further discloses the step of sending as having the same limitations as claim 36; therefore, the same rejection used in claim 36 still applies here.

As per claim 59, and with respect to the computer system of claim 56 above, this claim includes the same step limitations as in claim 4. These limitations are rejected using the combination of Ishizuka and Woolsey as set forth in claim 4 above.

As per claim 60, and in reference to claim 59 above, this claim includes the same step limitations as in claim 5. These limitations are rejected using the combination of Ishizuka and Woolsey as set forth in claim 5 above.

As per claims 61, 65, 66, and 68, in reference to the system of claim 56 above, Ishizuka further discloses the step of sending, or compiling as having the same limitations as set forth in claims 39, 43, 44, and 46, respectively; therefore, the same rejections used in claims 39, 43, 44, and 46 still apply.

As per claim 67, and in reference to claim 66, respectively, from above, Ishizuka does not disclose the same step limitations as recited in claim 15. But these are addressed using the rationale from the combination of Ishizuka and Woolsey, as set forth in claim 15 above.

Response to Arguments

4. Applicant's arguments with respect to claim 1 as amended have been considered but are moot in view of the new ground(s) of rejection.

Art Unit: 2124

As mentioned by Applicants (Appl. Rmrks, pg. 11-12), Woolsey does not teach or suggest establishing a secure link and receipt policy used to determine a trusted source. The fact that Woolsey provide utilities that automatically check trustworthiness of source providers is construed as establishing a policy at the level of receiving outside data into a private system because a policy amounts inter alia to rules/procedures established by a business to enforce something to protect the integrity of the business resources; and Woolsey has just done that. The claim does not provide specifics as to clearly set forth how such policy for receipt would be different from the practices/means being employed in Woolsey's method of verifying. Establishing secure link encompasses knowledge of the sending and receiving end in order to preclude unwanted infringement into properties or attacks to business resources as mentioned above. Therefore, Woolsey's method encompasses such concept of securing that no such attacks be possible when outside data/code is received. The claims does not describe in unequivocal terms as to how such secure link would distinguish over Woolsey's method to secure incoming data from originating from mistrusted or un-secure sources.

Because the scope of the independent claims has changed as a result of the amendments, the rejection now necessitates further grounds in order to respond to those changes, even though Woolsey's teachings strongly implies if not already teach the subject matter coming from the added limitations. Since the claim scope has changed, new grounds are needed.

Conclusion

5. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

Art Unit: 2124

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Tuan A Vu whose telephone number is (272) 272-3735. The examiner can normally be reached on 8AM-4:30PM/Mon-Fri.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kakali Chaki can be reached on (571)272-3719.

The fax phone number for the organization where this application or proceeding is assigned is (571) 273-3735 (for non-official correspondence – please consult Examiner before using) or 703-872-9306 (for official correspondence) or redirected to customer service at 571-272-3609.

Art Unit: 2124

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

VAT

December 17, 2004

Kakali Chaki
KAKALI CHAKI
SUPERVISOR PATENT EXAMINER
TECHNOLOGY CENTER 2100